

## RELEASE OF Z80 RTL TO THE PUBLIC DOMAIN

Bob Buege

RTL Programming Aids  
10844 Deerwood SE  
Lowell, MI 49331

Over the past several years I have developed a variant of Forth which I call RTL for Relocatable Threaded Language. Unfortunately, the costs involved in marketing and supporting RTL have proven to be greater than the income generated and I have now largely abandoned the sale of RTL as a source of income, concentrating instead on application software. As a result, I am now releasing the Z80 version of RTL to the public domain.

The differences between RTL and Forth are far too extensive to describe fully in this paper. However the editing procedure must be described since the user will have to modify certain terminal specific words before the editor will become operational.

Screens are not supported by RTL and no source code is ever saved. Instead, a decompiler is used to regenerate source code as it is needed. As an example, let's consider changing the prompt symbol from ">" to the traditional Forth prompt "ok:". Tracing through various words will eventually show that the word "PROMPT" is responsible for making various checks on the integrity of the system and then displaying the prompt message.

To begin editing, the command "EDIT PROMPT" is given, causing the computer to clear the screen and generate a source code listing of the word "PROMPT". The cursor control keys are then used to place the cursor at the greater than symbol, which is deleted by pressing the delete key. Next the message "ok:" is typed in, finishing the new definition.

Once the definition is satisfactory, type control-Z to exit the editor and compile the definition. If an error is discovered while a word is being compiled, the computer will return to the edit mode and place the cursor at the start of the word which caused the error. If compilation is successful, the computer will recognize the fact that the new word has the same name as a previously defined word and will ask whether or not to replace the old definition. A yes response will cause the new definition to be assigned to the old "PROMPT" and the new word will not be added to the dictionary. In this case the prompt will immediately change to "ok:". A no response will add the new word to the dictionary, allowing it to be tested carefully before changing the old word and deleting the new one with the command "REPLACE PROMPT".

The problem with RTL as it is supplied is that the cursor control which is necessary for the editor to operate is very terminal dependent. The following words must be modified without the aid of the editor before the editor may be used:

CURSOR-UP	Move the cursor up one line.
CURSOR-DOWN	Move the cursor down one line.
CURSOR-LEFT	Move the cursor left one column.
CURSOR-RIGHT	Move the cursor right one column.
CLS	Clear the screen.
TOX	Move the cursor to column X.
TOY	Move the cursor to line Y.
TOXY	Move the cursor to column X, line Y.
DO-KEY	See text.

The necessary modifications to the first five words should be obvious from an examination of their current definitions as shown at the end of this paper. The variables LINE#A and COLUMNA are used to keep track of the cursor position at port A (the terminal). Similar variables (LINE#B and COLUMNB) are used to keep track of the position on a page when characters are sent to port B (the printer). This allows the computer to take care of margins, page numbering, tabs, etc.

The next three words are generalized versions of the cursor control words just described and are currently defined in terms of the more primitive cursor control words. These words don't have to be modified at this time, but the performance of the editor will be increased if they are changed to take advantage of cursor commands which are supported by your terminal.

The word "DO-KEY" is used by the editor to determine what action to take when a key is typed. Printable characters are inserted into text, the delete key deletes a character, control keys perform a variety of functions, and the arrow keys move the cursor.

The character or characters received by the computer when one of the arrow keys is pressed varies with the computer or terminal manufacturer and the user must verify that the word "DO-KEY" interprets the arrow keys correctly. The version of "DO-KEY" supplied by me and shown at the end of this paper expects the arrow keys to send the hexadecimal codes 8, A, B, and C.

Each of the above words must be defined correctly by manually typing in the definition and then replacing the old definition as was described in the example above. When all the modifications have been made, type "CPM" to exit RTL. The computer will then type the version number (which is incremented each time the user exits from RTL) and will ask for a file name to use in saving the updated version of RTL. I generally use the file name "RTLxxx", where xxx is the version number which was just supplied by the computer. The computer will add the suffix ".COM" before saving RTL.

I am willing to supply RTL on an eight inch SS/SD CPM disk at a cost of \$5.00 (\$10.00 for overseas orders) but I don't have the resources to supply the wide variety of 5 inch formats. Users are asked to help in the distribution through the use of bulletin boards.

```

: CURSOR-UP
  B EMITA% -1 +TO LINE#A
:

: CURSOR-DOWN
  A EMITA% 1 +TO LINE#A
:

: CURSOR-LEFT
  8 EMITA% -1 +TO COLUMNA
:

: CURSOR-RIGHT
  C EMITA% 1 +TO COLUMNA
:

: CLS
  1A EMITA% 1 TO COLUMNA 1 TO LINE#A
:

: DO-KEY
  7F EQ-
  IF
    DROP DEL EXIT
  THEN
  20 GE-
  IF
    INSERT EXIT
  THEN
  CASE
    CNT-@ CNT-A CNT-B CNT-C CNT-D CNT-E CNT-F CNT-G
    LEFT CNT-I DOWN UP RIGHT INSERT-CR CNT-N CNT-O
    CNT-P CNT-Q CNT-R CNT-S CNT-T CNT-U CNT-V CNT-W
    CNT-X CNT-Y CNT-Z ESCAPE
  END-CASE
:

```

TERMINAL SPECIFIC DEFINITIONS